

# Large Scale Finite Element Modeling Using Scalable Parallel Processing

*Tom Cwik\*, Daniel Katz<sup>†</sup>, Cinzia Zuffada, Vahraz Jamnejad*

Jet Propulsion Laboratory  
California Institute of Technology  
Pasadena, CA 91109

<sup>†</sup>Cray Research Inc.  
El Segundo, CA 90245

Finite element modeling has proven useful for accurately simulating scattered or radiated electromagnetic fields from complex three-dimensional objects whose geometry varies on the scale of a fraction of an electrical wavelength. An unstructured finite element model of realistic objects leads to a large, sparse, system of equations that needs to be solved efficiently with regard to machine memory and execution time. Different solvers can be used to produce solutions to these systems of equations including both factorization methods and iterative ones. Factorization methods lead to high memory requirements that limit the electrical problem size of three-dimensional objects that can be modeled. An iterative solver can be used to efficiently solve the system without excessive memory use, and in a minimal amount of time if the convergence rate is controlled,

In this work an iterative solver for use with our finite element codes was developed for the Cray T3D massively parallel processor located at the Jet Propulsion Laboratory. This development was completed in two stages. First, a matrix decomposition algorithm was constructed, properly decomposing the sparse *matrix entries* into data sets that were read by the T3D processing elements (PEs). It is noted that this is a different strategy than the usual *mesh* decomposition algorithms developed in the past. The second stage was the construction of an scalable iterative solver on the T3D that efficiently computes a solution of the sparse system. The initial sparse matrix decomposition algorithm was originally developed for a YMP processor and then ported to the T3D.

In this talk we will present an overview of sparse matrix methods for distributed memory machines, as well as the specific implementation of the iterative solver. Example solutions have been obtained for systems with over one-half million unknowns.